

B003725 Intelligenza Artificiale (2019/20)

Studente: Davide Pucci (6283923) — <2019-12-17 Tue>

Elaborato assegnato per l'esame finale

Istruzioni generali

Il lavoro svolto sarà oggetto di discussione durante l'esame orale e dovrà essere sottomesso per email due giorni prima dell'esame, includendo:

1. Sorgenti e materiale sviluppato in autonomia (non includere eventuali datasets reperibili online, per i quali basta fornire un link);
2. Un file README che spieghi:
 - come usare il codice per riprodurre i risultati sottomessi
 - se vi sono parti del lavoro riprese da altre fonti (che dovranno essere **opportunamente citate**);
3. Una breve relazione (massimo 4 pagine in formato pdf) che descriva il lavoro ed i risultati sperimentali. Non è necessario ripetere in dettaglio i contenuti del libro di testo o di eventuali articoli, è invece necessario che vengano fornite informazioni sufficienti a *riprodurre* i risultati riportati.

La sottomissione va effettuata preferibilmente come link ad un repository **pubblico** su [github](#), [gitlab](#), o [bitbucket](#). In alternativa è accettabile allegare all'email un singolo file zip; in questo caso è **importante evitare di sottomettere files eseguibili** (inclusi files .jar o .class generati da Java), al fine di evitare il filtraggio automatico da parte del software antispam di ateneo!

Identificazione di scribi in testi antichi

In questo esercizio si utilizzano implementazioni disponibili di Decision Tree (p.es. [scikit-learn](#) in Python o [Weka](#) in Java) al problema dell'identificazione dello scriba su testi antichi, come descritto in [De Stefano et al. 2018](#), cercando di riprodurre risultati analoghi a quelli riportati nella tabella 7 dell'articolo (limitatamente alla colonna DT), usando 10-fold cross validation per determinare l'iperparametro N_l e tralasciando il post-pruning (i risultati potranno quindi differire leggermente da quelli riportati nell'articolo). Il dataset è disponibile qui: <http://archive.ics.uci.edu/ml/datasets/Avila> È accettabile utilizzare un sottoinsieme dei dati se le risorse di calcolo disponibili sono insufficienti.